

**ISUTC INSTITUTO SUPERIOR DE
TRANSPORTES E COMUNICAÇÕES**

Programação 1



Sumário:

- Plano da disciplina,
- Plano de avaliação,
- Objectivos,
- Introdução a Java.

Plano da disciplina

- Introdução a Disciplina
- Fundamentos da linguagem Java
- Elementos da linguagem de programação Java

Plano de avaliação

Semanas	Disciplina		Avaliação	Pontos
	Periodo			
1	29-Jul-24	- 3-Aug-24		
2	5-Aug-24	- 10-Aug-24		
3	12-Aug-24	- 17-Aug-24	TP1	50
4	19-Aug-24	- 24-Aug-24		
5	26-Aug-24	- 31-Aug-24	mT1	100
6	2-Sep-24	- 7-Sep-24		
7	9-Sep-24	- 14-Sep-24		
8	16-Sep-24	- 21-Sep-24	T1	200
9	23-Sep-24	- 28-Sep-24		
10	30-Sep-24	- 5-Oct-24	TP2	50
11	7-Oct-24	- 12-Oct-24		
12	14-Oct-24	- 19-Oct-24	mT2	100
13	21-Oct-24	- 26-Oct-24		
14	28-Oct-24	- 2-Nov-24	T2	200
15	4-Nov-24	- 9-Nov-24	Div	150
Outras Avaliações				
Assiduidade mínima (%)			75%	
Semestre II				850
Exame				450

Objectivos da disciplina

- Representar em forma de algoritmos problemas baseados em factos reais;
- Configurar e manter um ambiente de desenvolvimento em JAVA;
- Descrever os fundamentos básicos da linguagem de programação JAVA, apoiados no padrão “Javadoc API”;
- Criar programas utilizando os elementos básicos da linguagem de programação em JAVA;
- Criar programas simples utilizando classes e objectos em JAVA.

Introdução à Programação em Java

➤ O que é programar?

Programar é orientar a máquina sobre como proceder em uma determinada situação.

➤ O que é um programador?

Programador (desenvolvedor) é um profissional em TIC's que escreve códigos, comandos e instruções utilizando linguagens de programação para desenvolver diversos tipos de programas e *softwares*.

Introdução à Programação

➤ Quais são as habilidades do programador?

- Entender,
- Analisar,
- Codificar,
- Implantar,
- Auto-Ensino.

Introdução à Programação

- **O que é linguagem de programação?**
- Def1: É nada mais que uma linguagem, escrita por humanos, que também é entendida por computadores.
- Def2: É uma linguagem formal que, através de uma série de instruções, permite que um programador escreva um conjunto de ordens, acções consecutivas, dados e algoritmos para criar programas que controlam o comportamento físico e lógico de uma máquina.



Introdução à Programação

➤ Tipos de linguagem de programação

A linguagem de programação é a base para a construção de todos aplicativos digitais usados dia a dia e são classificadas em dois tipos principais:

- Linguagem de Baixo Nível, e
- Linguagem de Alto Nível.

Introdução à Programação

➤ Tipos de linguagem de programação

- **Linguagem de Baixo Nível:** Além do facto do computador necessitar que lhe instrua com acções bem específicas, estas acções devem ser passadas para o computador numa linguagem que ele possa entendê-las.

➤ Linguagem Máquina

É utilizada para comandar directamente as acções do computador. As instruções são constituídas por uma sequência de dois símbolos discretos que são 1 e 0. É uma linguagem difícil de usar e de compreender por humanos e varia de computador em computador.

Introdução à Programação

➤ Tipos de linguagem de programação

- **Linguagem de Baixo Nível:** Além do facto de o computador necessitar que lhe instruem com acções bem específicas, estas acções devem ser passadas para o computador numa linguagem que ele possa entendê-las.

➤ Linguagem *Assembly*

É uma linguagem semelhante a Linguagem máquina, diferindo desta no sentido em que usa nomes simbólicos com significado para humanos em lugar de 1 e 0. Tal como a Linguagem Máquina, a Linguagem *Assembly* varia de computador a computador.

Linguagem Máquina

Machine Code

```
10011101000110100000
01100011010001110110
10000010111101101110
11110110001011011000
10000010011100011011
10010011000111000000
```

Linguagem Assembly

```
; Hello World for Intel Assembler (MSDOS)

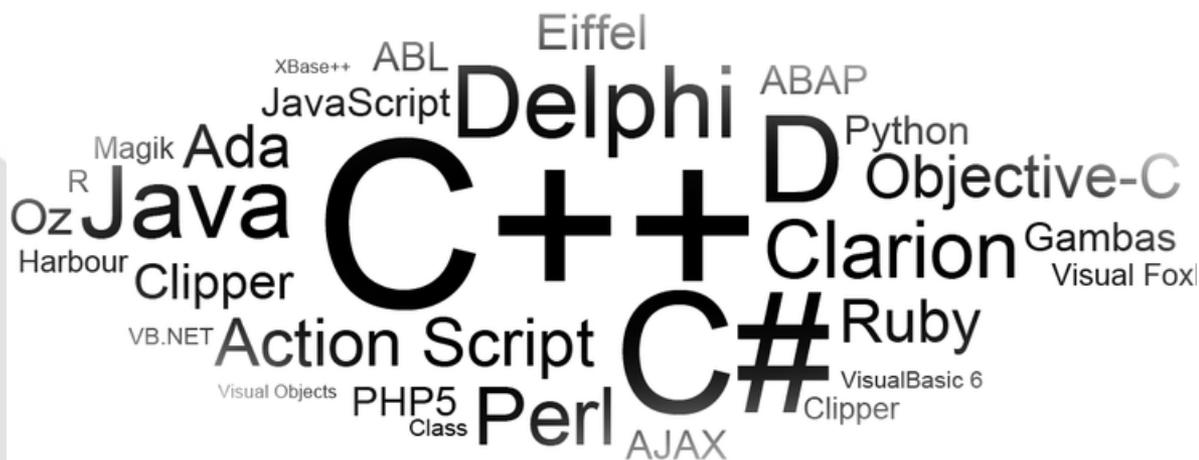
mov ax,cs
mov ds,ax
mov ah,9
mov dx, offset Hello
int 21h
xor ax,ax
int 21h

Hello:
    db "Hello World!",13,10,"$"
```

Introdução à Programação

➤ Tipos de linguagem de programação

Linguagem de Alto Nível: É uma linguagem que se aproxima das linguagens que os humanos usam para resolver problemas. É muito mais fácil de usar do que as linguagens máquina e *assembly*. Podem ser utilizadas em computadores diferentes.





Introdução à Programação

➤ O que é um programa de aplicação?

Programa de aplicação é um tipo de programa concebido para facilitar ao usuário a desempenhar determinados trabalhos.

➤ O que é um sistema operativo?

Permite-nos interagir com o computador assim como permite a interacção do computador com o exterior, é onde são implementadas as linguagens de programação de alto nível.



Introdução à Programação Java

➤ Java: definição

Java é uma linguagem de programação orientada a objectos e também Plataforma de Computação desenvolvida pela Sun Microsystems em 1995, usada para o desenvolvimento de sites, aplicativos e serviços.

➤ **ByteCode** é um código binário universal, isto é, é um código que pode ser executado em qualquer plataforma.

Introdução à Programação Java

- **JDK (*Java Development Kit*):** é o kit de desenvolvimento Java responsável por compilar código-fonte (.java) em bytecode (.class)
- **JVM (*Java Virtual Machine*):** é a máquina virtual do Java, responsável por executar o bytecode (.class).
- **JRE (*Java Runtime Environment*):** ambiente de execução do Java que fornece bibliotecas padrões do Java para o JDK compilar o seu código e para a JVM executar o seu programa.

Introdução à Programação Java:

Tradução

➤ Interpretação

Com efeito, cada vez que alguém deseja executar o programa, activa-se o interpretador, que traduz "no momento" (*on-the-fly* em inglês), cada instrução da linguagem alta, para várias instruções mais primitivas na linguagem baixa.

➤ Compilação

Vantagens: o compilador ajuda na detecção cedo de erros no programa, reduzindo o número de erros que aparecem no momento de execução. Também, como a maioria do trabalho da tradução é feito pelo compilador, antes do momento de execução, tais programas em geral executam mais rápido do que programas interpretados.

Introdução à Programação Java:

Tradução

➤ Compilação

Desvantagens: sempre, após alterar o código fonte (na linguagem "alta"), é necessário *recompila-lo* antes de poder ver os efeitos da alteração. Isto pode causar mais trabalho para o programador, mas na prática tem muitas ferramentas modernas que agilizam este processo.

Java é meio complicado pois é uma linguagem que usa ambas **compilação e interpretação**. (C# copiou esta ideia).



```
public class Hello{  
    public static void main(String args []){  
        System.out.print("Bem vindos a Programação I");  
    }  
}
```

Introdução à Programação Java:

A realizar no cmd do windows

➤ Compilação em java

javac Hello.java: produz Hello.class (código baixo intermediário).

➤ Interpretação em java

java Hello: interpreta e executa o programa



Introdução à Programação Java:

Programa

Um programa é nada mais que uma sequência de instruções, numa dada linguagem de programação, que mandam o computador fazer uma certa tarefa.

- A palavra técnica em inglês para uma instrução completa, que usaremos aqui, é o “***statement***”. Tipicamente teremos um ***statement*** para cada linha de um programa.
- Além disto, teremos algumas **linhas em branco** (para ajudar o leitor separar as partes do programa);
- Algumas **linhas de comentário** (notas numa linguagem humana, que o computador ignora, mas que acompanham o código de modo a ajudar outros programadores a entendê-lo).

```
1
2  /* comentario 1
3     comentario 1
4  */
5  public class Hello{
6
7     public static void main(String args []) {
8         // comentario 2
9         System.out.print("Bem vindos a Programação I");
10    }
11 }
12
```

linha em branco

Linha de comentário



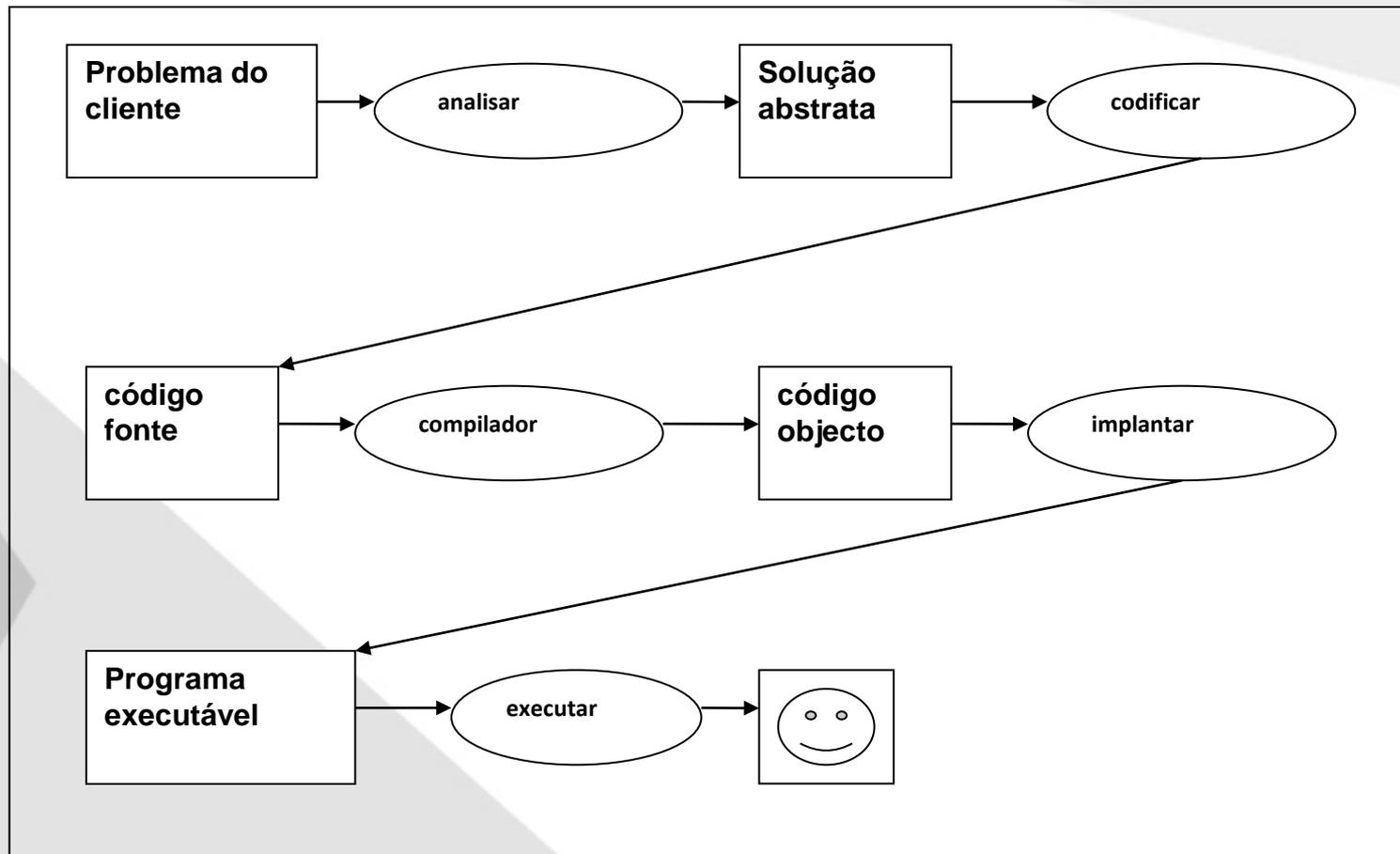
Introdução à Programação Java:

Os *softwares* geralmente têm muitas linhas de código mas podem ser decompostos em *módulos*, *componentes* ou *sub-sistemas*). E qualquer programa pequeno é composto de *statements*. Na verdade, somente existem cinco tipos importantes de *statements* no mundo:

- ***input***: que recebem informações de algures fora do programa (o teclado, a rede, um ficheiro ou base de dados)
- ***output***: que emitem informações criados pelo programa para algures fora (inclusive a tela!)

Introdução à Programação Java:

- **Instruções matemáticas:** que executam cálculos de aritmética etc. (ex. somar, multiplicar)
- **controle:** que verificam o estado de certas informações dentro do programa, e decidem qual o próximo *statement* a executar.
- **repetição:** faz uma certa acção (sequência de *statements*) repetidamente



Introdução à Programação Java:

Num mundo perfeito, cada programador ia percorrer o diagrama acima uma única vez para cada *job*, produzindo sempre no final, um programa perfeito que faz exactamente o que o cliente (ou chefe!) pediu. Mas como sabemos todos nós, o mundo é bem longe de ser perfeito! A verdade é que a primeira tentativa de fazer qualquer programa será cheia de erros. O termo técnico para tais erros é *bugs* (bichinhos).

É uma parte fundamental do trabalho de programador, a detecção, análise e resolução de *bugs*.

Introdução à Programação Java:

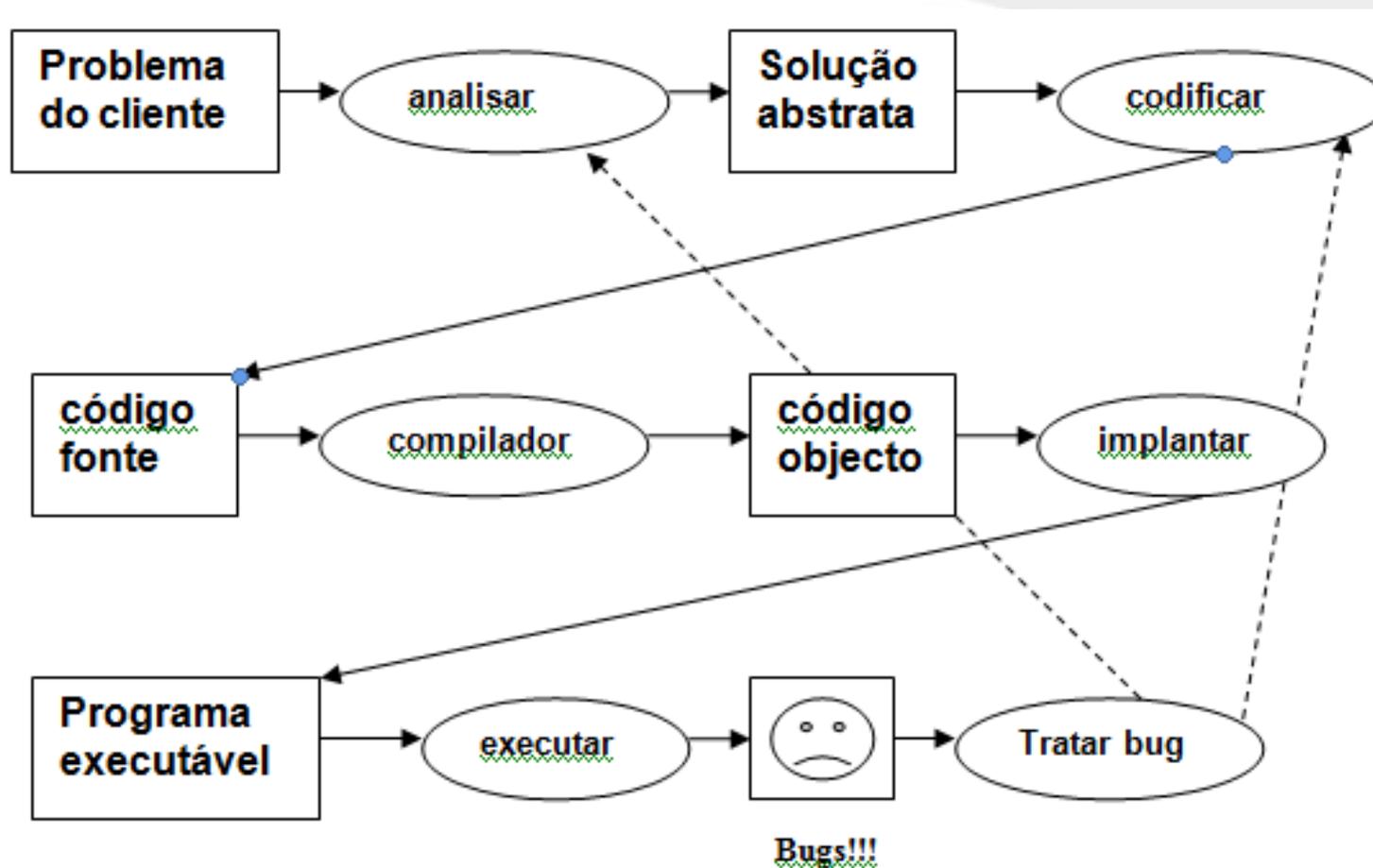
- **Bug de *software*** é um erro ou falha que ocorre num sistema ou programa de computador, resultando num comportamento incorrecto, inesperado ou fora do que tenha sido pretendido pelo desenvolvedor.
- **Debugging ou Depuração** é o nome dado ao processo de encontrar e remover os erros que podem acontecer num sistema ou programa de computador (*softwares*) bem como no *hardware*.

Introdução à Programação Java:

- "um verdadeiro programador nunca foge de *bugs*"
- "outras pessoas reclamam de *bugs* - mas o programador os concerta"

Possíveis erros (possíveis bugs):

- Erro de compilação
- Erro de *runtime* (execução)
- Erro lógico.



GARANTE O TEU FUTURO
COM UMA FORMAÇÃO SÓLIDA



Prolong. da Av. Kim Il Sung (IFT/TDM) Edifício
D1
Maputo, Moçambique

www.facebook.com/isutc

www.transcom.co.mz/isutc